AGENT-BASED SYSTEMS, THEIR ARCHITECTURE AND TECHNOLOGIES FROM LOGISTICS PERSPECTIVE

AĢENTOS SAKŅOTAS SISTĒMAS, TO ARHITEKTŪRA UN TEHNOLOĢIJAS LOĢISTIKĀ

V. Graudina, J. Grundspenkis

Holonic agent, holon, multi-agent system, multi-multi-agent system

1. Introduction

In a transportation and logistics domain a great number of actors are operating. These actors constitute supply chains, i.e. networks of suppliers, factories, warehouses, terminals, distribution centres, retailers and customers as well as various means of transportation, like ships, barges, trucks, trailers, vans, planes, etc. A logistics domain is decentralized, dynamic, where goals, organizations' capabilities and beliefs are continuously changing, and open, where organizations may enter or leave the system at any time [1].

Wide-ranging problems has been defined, part of which already solved, while others are in focus at the moment. Analysis of the great number of publications reveals that different mathematical modelling and simulation methods and techniques dominate for problem solving in transportation and logistics. At the same time new approaches start to appear, for example, Web-based, knowledge-based, intelligent agent and multi-agent based, mobile, etc. In this paper we focus on agent-based technologies and multi-agent systems (MAS) as one of the most perspective directions for the development of Web- and knowledge-based systems in transportation and logistics. The paper is based on the overview prepared as a deliverable of the EC 6th Framework project "Web-based and mobile solutions for collaborative work environment with logistics and maritime applications" (project acronym eLOGMAR-M). The purpose of the work at the project is to analyse different methods and techniques used for problem solving in transportation and logistics, in particular, advanced technologies, such as Web-based, knowledge-based, mobile, etc.

In transportation and logistics domain agent technologies emerge only recently in connection with such relevant, hard and, wherewith unsolved, problems as coordination in supply chain management, dispatching of transportation orders, efficient management of a seaport, or more specifically, a container terminal, etc. Intelligent agents represent the organizations within the logistics domain, and model their logistics functions, processes, expertise, and interactions with other involved organizations [1]. Some agents simulate customers; others are elements of the traffic infrastructure, or transportation units (trucks, trains, planes, ships, etc.) [2].

Multi-agent systems are highly applicable in domains and problems where centralized approaches meet their limits because multi-agent systems have such features as parallelism, robustness and scalability. Moreover, multi-agent based approaches suit for domains, which require integration and interaction of multiple actors and knowledge sources, the resolution interest and goal conflicts, or time bounded processing of data [3]. Therefore, these approaches allow distributed task modelling that make them very attractive for supply chain management which has become a dynamic process that involves the simultaneous acquisition and continuous reevaluation of partners, technologies, and organization structures [4]. The maturity of technical foundations for multi-agent systems and the support by the development tools and methodologies leads to increasing number of multi-agent systems proposed for supply chain management [4, 5, 6, 7, 8]. It is worth to underline that agent-based systems and technologies are much more popular in supply chain management than in other directions of logistics.

The objective of this paper is to give an insight into agent-based systems, their architectures and technologies that already developed for the logistics domain. The remaining paper is organized as follows. Some logistics problem solutions on the basis of agents are highlighted in Section 2. A new agent technology, so called, holonic agent is discussed in Section 3. Multi-multi-agent systems that recently appear in supply chain management are analysed in Section 4. Different agents for logistics domain are reviewed in Section 5. Finally, the conclusions are made and future work is outlined in the last section.

2. Some solved transportation and logistics problems using agent- based technologies

Analysis of different information sources allows to conclude that applications of intelligent systems, in general, and agent-based technologies, in particular, cover such problems as traffic modelling, logistics planning, sea freight transportation, vehicle dispatching, transportation scheduling, supply chain management, management of container terminal systems, decision support for letter transportation, and others [1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]. The agent organization is compliant to a socio-technical system, which improves the understanding of decisions taken by agents for actors. These organizing principles are generic for any logistics system and allow fast adaptation to changing conditions.

Intelligent systems, which are able to assist the design-phase (strategic planning) of traffic and transportation systems and/or the management phase (tactical and operational planning) as well, are of special interest in transportation and logistics. The purpose of logistics is to design, to organize and to manage transportation in order to meet customer service demands, cost and environmental requirements. In the field of transportation logistics the focus is on the analysis of urban, regional and intercity transportation networks for both passenger and freight transportation as well. Complex *hybrid – type systems*, e.g., intermodal transportation systems, which include air-, road- and rail transportation are of particular interest [16].

Intelligent systems are designed for real world applications in traffic and transportation. They are built on the basis of an advanced software engineering concepts including object-oriented software development and integration with non-standard databases and GIS. Several so-called intelligent techniques coming from artificial intelligence, operation research and computational intelligence, such as evolutionary and genetic algorithms, constraint programming, agent-based modelling and simulation, tabu search metaheuristics and high performance optimisation and simulation techniques are used on the algorithmic side [16]. Models such as modal split models, models of transportation network design, models of vehicle routing and crew scheduling, models for the estimation of future volume of traffic, etc. are developed on the modelling side [16]. The plethora of techniques for intelligent systems in transportation and logistics is shown in Figure 1.

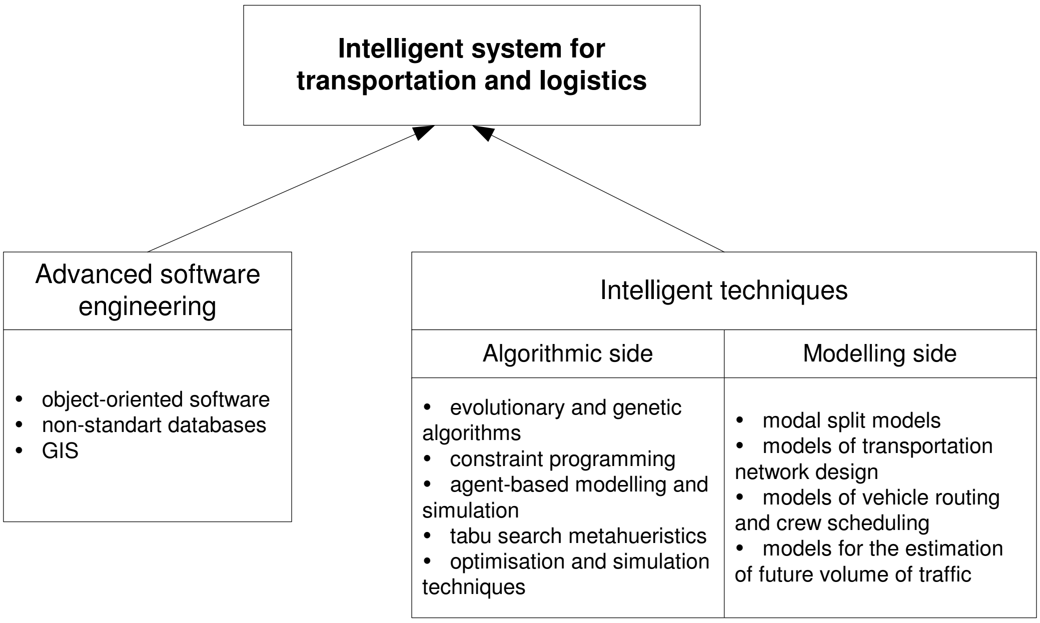

Figure 1. Basis of the intelligent systems for transportation and logistics

Computer-based simulation can provide the decision-makers with help they need in creating the strategies for the development of intermodal terminals [10]. In situation of hard competition there is a need to enhance the infrastructure of intermodal terminals to make them more attractive for customers. At the same time very often there are substantial costs associated with interactions of different transport modes that, in their turn, dictate a need for trade-offs. So, well-designed investment plans must be constructed. Simulation has proven to be a reliable and convenient tool to support the decision makers in the daily operations in many cases [17, 18, 19]. A substantial help to terminal managers can be derives from decision support systems (DSSs) where planning and management techniques, derived from the operation research and artificial intelligence fields, can be coupled with simulation models and statistical data analysis tools. A well-designed simulation tools can be the middle ground where decision makers compare their own experience with DSS generated management policies and validate them. Major problems in an intermodal container terminal are the following: storing containers on the yard, allocating resources in the terminal and scheduling train, vessel or truck loading and unloading operations. To solve these problems an architecture has been defined that consists of three different but strictly connected modules [20]: (1) a simulation model of the terminal, (2) a set of forecasting models to analyse historical data and to predict future events, (3) a planning system to optimise loading/unloading operations, resource allocation, and container locations on the yard. This architecture supports the terminal managers in the evaluation of [11]: 1) vessels loading and unloading sequences in terms of time and costs, 2) resource allocations procedures 3) policies for container storage both in terms of space and costs of operations. Simulation tool is based on the partition of simulation objects between simulation agents and simulation components.

The use of multi-agents assisting managers offers possibilities to manage a seaport, or more specifically, a container terminal more efficiently [9]. The managing of ports is complex and crucial to the efficient and effective operations that are increasingly demanding due to the fact that modern ports are no longer passive points of the interface between sea and land transport, used by ships and cargo as the point of intermodal interchange. Instead, they have become logistics centres in the global transport systems. The coordination of information and the physical flows of cargo in seaports has been a problem for a long time because various actors (customers, freight forwarders, ship agents, stevedores, and port authorities) are involved. In this decentralized problem solving area seaports as complex systems, where various organizations and actors interact, can be modelled as agents in multi-agent systems (MAS). The MAS approach allows each agent to communicate through the array of networks and systems that make up the container terminal system. In [9] a computer model of a container terminal in order to conduct simulation of its management is proposed. The managers of the container terminal are the ship planner, port captain, yard planner, and stevedore, and, they are mapped on corresponding agents of the MAS. The CommonKADS is used to model the knowledge of the managers in the container terminal.

The agent-based framework may lead to more effective integration of production, logistics, and trading processes. *Integrated commerce* (I-commerce) is an operational extension of traditional e-commerce that entails getting customers more involved in ordering activities so that contractors can more efficiently fulfil orders. I-commerce also entails more effective practical integration of supply-chain processes offering its users several I-commerce techniques (such as mobile timber auctions with integrated support of optimizing logistics) for negotiating, communicating, and exchanging information more effectively [21]. More detailed on new trends in supply chain management based on multi-agent technologies are highlighted in Section 4.

3. Technologies for problem solving in transportation and logistics

This section does not include the exhaustive analysis of different technologies already used. Instead, we focus only on the some of them, namely, on those, which are the most interesting from our viewpoint. During a design of agent-based systems for transport a new agent technology has been introduced. It is a holonic agent or a holon [22]. A holon is composed of agents working together in order to achieve a common goal. The system's users or the other members of the agent society can interact with a holon as if it is a single agent. This allows to model several levels of abstraction in a appropriate way. The term super-holon is used to denote a composition of subordinate agents, which are called sub-holons or sub-agents.

Holonic structures can be designed in different ways depending on organization of agent societies [23]:

1. <u>A holon as a federation of autonomous agents</u>, where sub-holons are fully autonomous agents and super-holon is just a new conceptual instantiation of the same agent architecture. No agent has to give up its autonomy, and the super-holon is realized exclusively through cooperation among the sub-holons. This holonic structure is just another way of looking at the traditional multi-agent system (see, figure 2 a).
2. <u>Several agents merge into one</u>, where a new holonic agent is created as union of all involved agents. Merged agents completely give up their autonomy but after super-holon is terminated agent may be reverted into their initial state (see, Figure 2b).
3. <u>A holon as a moderated group</u> is a hybrid way of forming a holon, where agents give up only part of their autonomy to the super-holon. The super-holon is one of agents which becomes as a representative or head of the holon. The head coordinates the resource allocation within the holon and controls the communication with the rest of the agent society. The competence of the head of a

holon range from pure administrative tasks to the authority to issue directives to the other agents. Agents included in holon keep their own plans, goals, and communication facilities in order to provide their resources for the transportation plans according to their role in the society (see, Figure 2c).

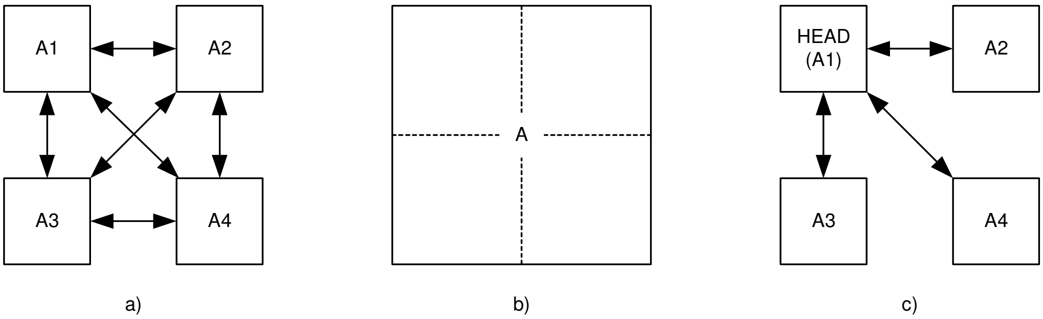

Figure 2. Holonic structures [adapted from 23]

In a multi-agent systems holonic structures occur when agents not only cooperate but have to be combined in order to perform their tasks [24]. Autonomous agents may join others and form holons without loosing their autonomy completely if they have the "freedom" to leave the holon again and act autonomously or rearrange themselves as new holons.

Holonic approach is suitable for solving problems in logistics and transportation, for instance, the fleet scheduling problem, because this domain comply with most of characteristics of holonic domain [23]:

- Operator abstraction. Holonic systems are well suited for domains with actions of different granularity. Macro-level actions are carried out by holon's head and decomposed onto the sub-holons. Actions can be defined at different levels of abstraction: the most general level specifies which transportation task a vehicle has to perform. These actions are then recursively decomposed into actions of lesser abstraction, such as loading, driving or vehicle maintenance, which again consists of sub-actions such as docking to a terminal, using traffic information systems, refueling, etc.
- Hierarchical structure. A domain that exhibits a hierarchical structure is usually an excellent candidate for a holonic system, since hierarchies of sub-holons can be modeled canonically. In transportation domain vehicles compose hierarchy by building up transportation units.
- Decomposability. Traditionally a pre-requisite for multi-agent systems is decentralized and decomposable problem setting. In this case pro-active and autonomous agents are assigned to corresponding sub-problems. Furthermore, due to the fact that holonic agents are structured hierarchically and realize actions of different granularity, holonic agent systems can naturally deal with decomposable problems.
- Communication. If the overall problem is decomposed into sub-problems, communication among problem solvers is needed. Sub-agents of a holon are communicative and hence, holons provide facilities for intra-holonic as well as inter-holonic communication. In fleet scheduling problem coordination among units of a common vehicle requires a lot of communication, cooperation among units of the same company that are not included in the same vehicle need less

communication, while units belonging to different companies do not communicate at all.

- Social elements. The setting is cooperative with a company and competitive between involved companies (in non-cooperative settings the use of holonic agents is not reasonable).
- Situatedness and real-time requirement. Some situations require a real-time behavior when the problem solver has to find a fast and real-time answer within limited computation time, e.g. in case of re-planning during execution time when urgent orders have to be scheduled immediately.

To summarize, the most important requirements for a holonic agent are structure and cooperation, i.e., the domain should have a holonic structure (should be recursively decomposable) and must have sufficient cooperative elements between the distinguished problem solvers. The important difference to traditional multi-agent agent domain in the possibility to model centralistic aspects of a domain [23].

This approach is successfully used for TELETRUCK system [23, 25]. The TELETRUCK agent society is implemented as a holonic agent system. The main unit in the system's architecture is the central module that is a holonic multi-agent system. The central module manages the planning and optimization of the vehicle configuration and tour plans. Both, the multi-agent system and the user (the dispatch officer) access a common SQL-database. The user introduces tasks to the multi-agent system and can generate tour plans by hand or using the multi-agent system. He also can modify plans generated by the system, can impose additional constraints and can request an optimization of the solution. The on-board computer of the truck receives the GPS information and updates its position in the database. Electronic maps and routing software are added to supply the tour planning systems with geographical data [23]. The TELETRUCK system was designed as an agent-based forwarding system, able to manage the business processes of forwarding companies. Whenever the system plans a road-based transport, the agents representing the involved physical components form a vehicle holon.

In TELETRUCK system the basic transportation units (trucks, trailers, drivers, chassis, and containers) are modelled as component agents. These agents merge into a holon that represents the vehicle for the transportation task. The vehicle holons are headed by PnEU *(Planning 'n' Execution Unit)*, a special agent that is equipped with planning capabilities. The vehicle holons and the agents representing currently idle transportation units form a super-holon that represents the whole transportation company. The head of company holon, called the company agent, coordinates the interaction with the user and communicates with other companies that employ the TELETRUCK system. The central module in TELETRUCK is a holonic agent system that manages the planning and optimization of the vehicle configuration and the tour plans. For each transportation unit of the forwarding company there is an agent that administrates its resources. These holonic agents have plans, goals, and communication facilities in order to provide their resources for the transportation plans. The agents can merge with a PnEU and form a holon that represents a complete vehicle.

TELETRUCK is an extension of the MAS-MARS system, a research prototype for transportation scheduling [26]. The agents of MAS-MARS represent homogeneous trucks. They negotiate on incoming orders and optimize the distribution and execution of these orders according to an abstract cost measure. In contrast to that TELETRUCK system is an application prototype developed in close collaboration with the forwarding company. It schedules realistic orders using heterogeneous agents modelling different forms of vehicle. A central idea underlying the TELETRUCK approach is to model the basic physical objects (drivers, trucks, trailers, containers) of the transportation domain explicitly by basic agents. These agents have to join together and form holonic agents that act in a corporate way.

The TELETRUCK project has an extension – TELETRUCK-CC [24] that allows several independent shipping companies to cooperatively optimise their fleet schedules. Later it has been used as a base for PLATFORM project [12], as it is shown in Figure 3.

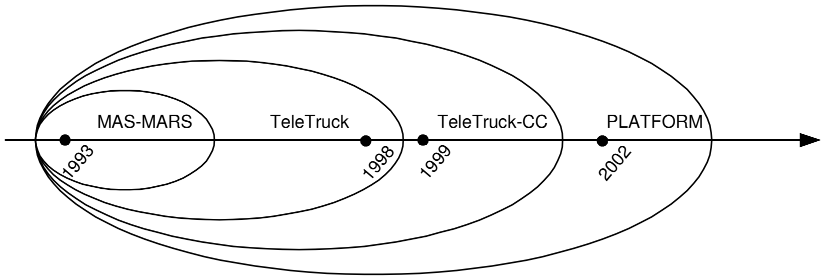

Figure 3. Evolution of systems based on holonic approach

The PLATFORM architecture consists of two subsystems [12]:

1) The intermodal transport planner that manages the planning of the whole intermodal transport chain from origin to destination for an intermodal transport unit. The intermodal transport planner plans the whole intermodal transport task thanks to: (1) intermodal planning and execution units (they contact specialized agents for planning, booking, reservation of the initial and final leg on the road and the main leg by train); (2) forwarding agents are responsible for planning of delivery intermodal transport units to and their pick-up from terminals (each forwarder is modelled by such a forwarding agent); (3) booking agent checks for availability of places on scheduled trains, checking which bookings are possible and then chooses the bet and makes the reservation.

2) The simulation system that simulates intermodal transport unit transport process, both assessing the feasibility of plans generated by the intermodal transport planner and evaluates the performances of intermodal terminals, based on a detailed description of the intra-terminal processes. The simulation system performs the execution of intermodal transport tasks including internal terminal operations to evaluate their feasibility and their performances. It is composed of: (1) road simulator, that simulates the transport of the intermodal transport unit by truck, as delegated to forwarders. It simulates the flow of incoming and departing trucks at each terminal in the corridor; (2) terminal simulator simulates the loading and unloading of intermodal transport units from trucks and trains as well as storing of intermodal transport units in the intermodal terminal; (3) train simulator simulates the flow of trains within the chosen rail corridor, according to the train time-tables, and the flow of trains from and to the terminals [12].

Another kind of holonic agent as a software agent is realized in the Casa ITN system [21]. Each member of participant groups (producers, buyers, retailers, and logistics companies) is represented with an appropriate holonic agent. Holonic agents accomplish complex (mostly hierarchically decomposed) tasks and resource allocations in the selected application scenarios. They also coordinate and control the activities and information flows of their subagents. This holonic technique lets personal assistants that represent human users act on behalf of their users even if those users are offline. A corporation holon consists of several holonic agents, each of which represents a special department and its corresponding tasks and services.

The holon agent approach can be successfully used also for solving problems like job-shop scheduling of work in a production plant, dynamically scheduling a production plans or coordination of business processes in a virtual enterprise.

4. Multi-multi-agent systems for supply chain management

Multi-agent systems (MAS) allow distributed task modelling that is needed for supply chain which involves many different interdependent tasks, i.e., planning execution and controlling of production, transportation and warehousing processes [6]. Thus different MAS specializing on certain tasks have to interact with each other. The successful integration of many MAS that perform both inter- and intra- organizational planning and execution tasks could lead to an improvement in the design of supply chain management systems [6]. Various MAS have been developed and applied due to the growing success o f FIPA standatization (http://www.fipa.org) for MAS, increasing availability of FIPA-compliant frameworks, such as JADE (http://jade.cselt.it) and the availability of an open service infrastructure, for instance AgentCities (http://www.agentcities.org) [5, 6].

Several methods exist, which focus on building a single MAS and support at least one of the established development phases (analysis, design, implementation and deployment), for example Gaia, PASSI, MASSIVE, MaSE, AUML, etc.

The Gaia methodology is developed for agent-oriented analysis and design and can be thought of as a process of increasingly detailed models of the system to be constructed [27]. The objective of the analysis phase is to understand the system and its structure. The system's organization is viewed as a collection of roles that have certain relationships to each other. The organization model in Gaia consists of the role model and the interaction model. The role model identifies the key roles in the system that are characterized by two types of attributes, namely, the responsibility of the role and the permissions/rights associated with the role. The interaction model, in its turn, represents relationships and dependences between the various roles in the MAS. The aim of the design phase is to transform the analysis models into a sufficiently low level of abstraction so that design techniques, e.g., object-oriented techniques may be used in order to implement agents. The Gaia design process generates three models: the agent model that identifies the agent types and instances, the service model that identifies the main services that are required to carry out the agents' roles, and the acquaintance model that documents the communication links between the different agents.

PASSI (Process for Agent Societies Specification and Implementation) is a step-by-step requirement-to-code methodology [28] PASSI consists of five models and twelve process steps for building MAS, supports analysis, design and implementation phases, uses the UML notation. At the analysis phase the system requirements model is built. UML use-case diagrams are used to describe the agent domain. The use-case diagrams help to identify the agents and their roles. The design phase is characterized by the Agent Society Model that consists of an ontology description, a role description and a protocol description. The advantage of the PASSI methodology is the ability to model the agents' knowledge and the communication specification as two related elements in the same way. The agent implementation model defines the resulting system architecture definition. A behaviour definition using activity diagrams is used to describe the MAS. All models have to be converted into a code model. The strength of PASSI is the integration of known object-oriented design methods into MAS design.

MASSIVE (Multi-agent Systems Iterative View Engineering) provides a framework for the development of MAS [29]. MASSIVE applies standard software engineering techniques and new software development approaches to MAS. This methodology is rather view-oriented than process-oriented. Views are used to describe different aspects of the complete design. The logical decomposition of a system contains seven views. The task view depicts the system from a functional perspective. The environment view reflects the system and the developer perspective. The role view describes the role model of agents by analysing the functional and physical relations within the system. The interaction view models the interaction needed for solving a problem. The society view takes a macro perspective for building up structured collections of MAS. The resulting software model for the MAS and the single agents are described in the architectural and the systems view. Interactive view engineering, i.e., a stepwise refinement of the process model where views are embedded is used in this approach. There are two advantages of MASSIVE methodology. First, the model can deal with an incomplete problem specification and is not fixed for the entire project lifetime. Second, MASSIVE provides a conceptual framework for enabling a systematic learning process that allows to impose the model according to the experience gained in the development process.

Looking on MAS from the supply chain management design viewpoint, these systems depend on each others' input and output, and therefore need to be integrated in heterogeneous systems. In [6] an approach for integration of MASs that leads to the concept of multi-multi agent systems (MMAS) is described. The structure of integrated MASs, i.e., MMAS, is designed to inherently meet the requirements of distributed supply chains where information for integrated production planning and control is not available within the whole supply chain. Integration of complex systems requires agreements of architectural and technical nature in order to avoid a time-consuming struggle with implementation details.

The analysis, design and implementation of MMAS has resulted in a system called *Agent.Enterprise* [5, 6, 30]. Unlike described above Gaia, PASSI and MASSIVE methodologies, the *Agent.Enterprise* methodology focuses on a distributed and weakly coupled development process. The results of the analysis and design phase are consolidated in functionally restricted prototypes, which constitute a test bed, for the components of the MMAS [6]. The project substitutes their prototypes with so called gateway-agents in order to connect their applications to the common scenario. Below the short description of the MMAS development process is given. More details, may be found in [3, 5]. The analysis phase includes Role Definition and Assignment step followed by the use case specification. The first step is the performance of role-playing technique to simulate the exchange of information between the systems (the projects' MASs). As a result the communication acts between the projects' MASs and the required information are specified informally and have to be formalized into a use case specification. For the design phase of *Agent.Enterprise* two central design decisions are united in the so called gateway-agent concept (see Figure 4.)

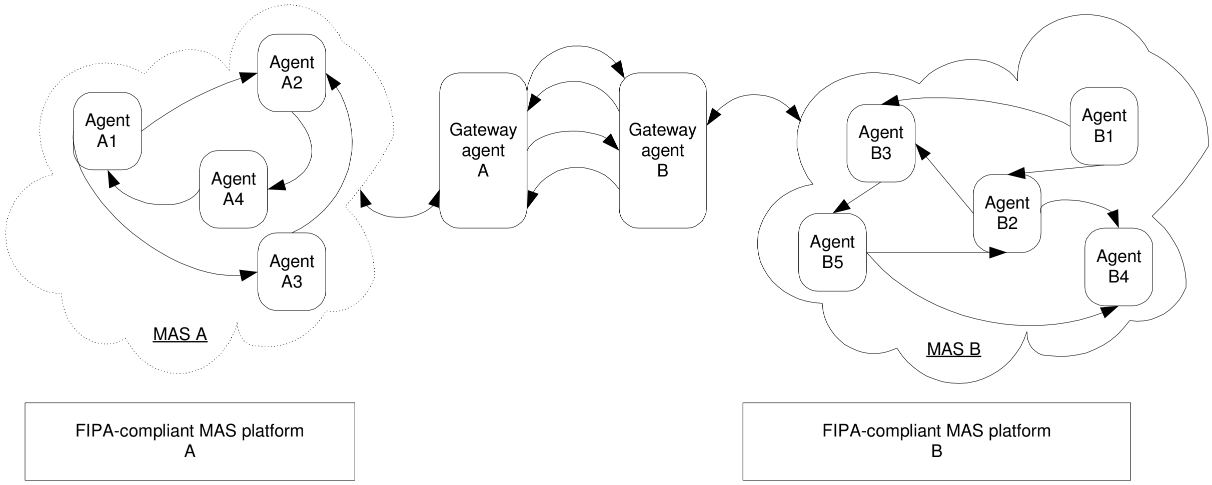

Figure 4. Multi-multi-agent system architecture [adapted from 6]

Firstly, the use of FIPA-compliant platforms avoids many of the communication-related obstacles and allows to concentrate on domain aspects. Secondly, every individual MAS to be integrated should be represented by a single agent, that comprises all roles of the corresponding MAS and provides them to the resulting MMAS. After the analysis phase where for the roles for each participating MAS are defined, the design phase starts with speech act design. The purpose of this phase is to overcome a language barrier for the communication due to the heterogeneity of knowledge representation and semantics in the individual systems. The gateway-agent concept defines a virtual MAS where agents are scattered across a number of agent platforms such that an ontology can specify the semantics of the conversations. Ontological expressions are used as a means of communication. The next step in the overall design process is to define dynamics in conversation, i.e., to make decision which interaction-protocol have to be used for communication. This is solved in interaction-protocol design step.

The implementation phase starts with the implementation of functionally restricted prototypes that serves for a consolidation of speech acts and interaction protocols. The implementation of these prototypes are closely bound to test-sessions. The gained experience is used for improvement and refinement of speech acts and/or interaction protocol design.

There are several advantages of the gateway-agent concept. Developers can focus on a single agent and only the gateway-agents must be available during operation. The *Agent.Enterprise* methodology addresses special needs, e.g. dealing with individual ontologies that arise from the nature of MAS integration. The methodology is designed to fit the need for distributed development.

The resulting MMAS called *Agent.Enterprise* covers services of supply chain scheduling, shop floor production planning and control, and pro-active tracking and tracing services. The MMAS provides reliability of overall supply chain processes and may be considered as an extension to MAS in the context of supply chain management [5].

5. Different agents for intelligent transportation and logistics systems

From a logistics management perspective agents taking responsibility in fulfilling the operational goals as well as agents supporting these tasks in a computational way are needed.

These considerations motivate the introduction of two *generic meta-types of agents* in the logistics domain. *Management agents* pursue goals with respect to their environment and their defined action space, whereas their contractors, the *service agents* solve well specified tasks autonomously [31].

Management agents are the central part of a management system. In logistics they are software entities for meeting operational goals on behalf of a human actor or another managerial agent with some degree of independence or autonomy, and in doing so, employ some knowledge or representation of user's goals or desires. Management agents are pro-active, goal directed, take goal responsibility, make decisions, and have a model of their environment. They act autonomously, but the actions are constrained by the provided information and models (e.g. from a *supervising agent*). A management agent needs the following information: a goal to pursue, its skills and behaviour, model of its environment, its role in the agent community (e.g. supervising agent, its sub-agents, *collaborating agents*), communication and co-operation protocols. In a logistics domain, where the co-ordination of joint actions play an important role, a management agent must exhibit the following properties: act in a collaborating manner, apply various problem solving strategies, manifest communicational abilities, use methods for solving conflicts among its sub-agents, and show capabilities for goal and model building, which are used by its sub-agents.

The design of DIAL system (Distributed Intelligent Agents for Logistics) is based on a multi-agent framework where a customer's problem can be decomposed and assigned to one or more (as needed) intelligent agents, which together generate a logistics plan [32]. Intelligent software agents are built on the top of simulation models to communicate among themselves and to generate or assist in generating a correct course of actions. These software agents must have sufficient reasoning power to understand each other's messages and actions as well as their own objectives in generating a complete plan.

The role of intelligent agents is as follows [13, 14]:

1) a *priority agent* prioritizes cargo based on unit's Latest-Arrival Date, the Available-to-Load Date at the port, and the size of the unit;
2) a *unit integrity agent* provides Integrated Computerized Deployment System with a list of items that meet the unit integrity rules;
3) an Integrated Computerized Deployment System *interface agent* passes Port Simulation Model a per-stow plan for use in staging area setups;
4) a *timing agent* passes Port Simulation Model packages of data in each 24 hour time period based on the outcome of Force Flow Model that provides time-phasing information for cargo from the installation to the port;
5) a Port Simulation Model *interface agent* passes Integrated Computerized Deployment System a list of items for the final stow plan that meet all the combined rules of unit integrity, Force Flow Model, and Port Simulation Model.

Previously mentioned kinds of agents are used in the DIAL project [33], which is an open system architecture that allows existing models to interface and communicate in a distributed network environment to evaluate and develop logistics plans.

In case of intermodal transport an intermodal terminal is represented by a *holonic terminal agent society* consisting of the *terminal agent*, which is the head of the society, a *booking agent* managing the booking requests for the trains handling in the terminal, and *locomotive agents* which represent the trains. Inter-connecting the two transport modes allowed for intermodal transport orders requires more sophisticated planning competences and execution processes [12]. In intermodal transports an *Intermodal Plan'n'Execute Unit* is introduced that plans and executes the plans for all the goals comprised within the order. The *Intermodal Planning and Negotiation Protocol* is an application-specific extension and nesting of several classical Contract Net Protocols [15, 34].

The overall quality of supply-chain management can be improved with more sophisticated planning, scheduling, and coordination methods. In recent years, new software architecture for managing the supply chain at the tactical and operational levels has emerged [8]. It views the supply chain as composed of a set of intelligent (software) agents, each responsible for one or more activities in the supply chain and each interacting with other agents in planning and executing their responsibilities. In typical agent decomposition there are also agents that are responsible for logistics and transportation. These agents are called *logistics agent* and *transport agent*, respectively.

- *Logistics agent* is responsible for coordinating the plants, suppliers, and distribution centers in the enterprise domain to achieve the best possible results in terms of the goals of the supply chain, including on-time delivery, cost minimization, and so forth. It manages the movement of products or materials across the supply chain from the supplier of raw materials to the customer of finished goods.
- *Transportation agent* is responsible for the assignment and scheduling of transportation resources to satisfy interplant movement requests specified by the logistics agent. It can consider a variety of transportation assets and transportation routes in the construction of its schedules.

Supply chain coordination can be analyzed from the point of view of a typical logistics scenario [7]. The scenario includes a variety of logistics service providers, manufacturers, and suppliers. The logistics coordinator acts as a function broker that is linked to multiple manufacturers and transporters who provide the manufacturing and logistics services. *Coordination agents* are designed to accomplish order coordination tasks in organizations. These agents are *Distribution Hub Agent*, *Logistics Coordinator Agent*, *Manufacturer Agent* and *Transporter Agent*. These four types of agents represent the generic roles in a typical supply chain.

An architecture for agent-based logistics coordination has been designed based on JADE agent platform (http://jade.tilab.com). Each participant of a supply chain provides the agent platform with a set of agent instances. The agent platforms are linked via Internet connections. These platforms include basic *management agents* and *application agents*. Basic management agents embody *Registration Agent*, *Communication Agent*, and *Directory Agent* to facilitate the creation and management of application agents. The *logistics management agent* is actually an instance of the application agent, which is used to coordinate and produce the optimized logistics decision. The application agent includes interface, activation controller, optimization/planning modules, and knowledge base. Different supply chain planning/optimization modules such as demand forecasting, and transportation/manufacturing planning could be incorporated into the platform.

On the other hand, in [8] it is proposed that typical agent decomposition for supply chain management is as follows. *Order acquisition agent* which is responsible for customers, their orders, and communication with *logistics agent* in cases of modification or cancelling of orders. *Logistics agent* coordinates and manages movement of goods across the supply chain. *Transportation agent* is responsible for transport resource scheduling. *Scheduling agent* is responsible for scheduling and rescheduling activities in the factory, and the resource assignment. *Resource agent* manages availability of resources. *Dispatching agent* performs the order release and real-time floor control functions.

During last years, the new kind of agents, that are, *mobile agents* are involved, which not only communicate and interact with other agents but also move from one place to another to find and process information, as well as to send back results to the users [35]. Mobile agents provide an innovative concept for distributed systems development. Global transportation networks and supply chains require distributed global planning of activities. So, it is easy to foresee that the role of mobile agents in transportation and logistics problem solving, and supply chain management in nearest future will grow very rapidly.

One of main agents' functions in logistics and transportation systems is collaboration support. This can be divided in the following subfunctions: collaborative information sharing, collaborative operation and collaborative configuration [4]. Main agents for that are *searching agent*, *scheduling agent*, *optimization agent*, *negotiation agent* and others.

One can conclude that at the moment a plethora of various agents already exists, many research activities are carried out, and application projects are running at different universities. New architectures, e.g., holons and multi-multi agent systems have emerged and methodologies for their development have been proposed more application areas in transportation and logistics, in particular, in supply chain management are covered by agent-based systems.

6. Conclusions and future work

This work is based on analysis of the great number of different information sources. As the result we conclude that various agents have been developed and applied in the transportation and logistics domain. Agent-based systems that mainly are used to solve typical problems in this domain are decision support systems, logistics planning systems, supply chain management systems, and simulation and modelling systems, which support both decision making and planning. The extension of single agent-based systems are holonic systems and multi-agent systems (MAS). The focus of the most of the methods for their development is on building single (most often closed) holons and MAS. In contrast, the methodology for multi-multi agent system (MMAS) is proposed as an open system architecture allowing to integrate MAS.

The future work is connected with collection of more information about specific agents and multi-agent systems and their applications in the transportation and logistics domain, especially in intermodal transportation. The more detailed analysis of existed approaches of applying agent technologies to automate the coordination and decision making, already proposed methodologies for single agent-based, MAS and MMAS development and the corresponding software environments and shells is needed.

7. ACKNOWLEDGEMENT

This work has been partly supported by the European Social Fund within the National Program "Support for the carrying out doctoral study program's and post-doctoral researches" project "Support for the development of doctoral studies at Riga Technical University".

This paper is based on research supported by the 6th Framework EC project No. 511285 eLOGMAR-M.

8. REFERENCES

1. Perugini, D., Wark S., Zschorn S., Lambert D., Sterling L., Pearce A. Agents in Logistics Planning-Experiences with the Coalition Agents Experiment Project. In Proceedings of workshop at the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2003). Melburne, Australia, July 2003.
2. Zhu, K., Bos A.. Agent-based design of intermodal freight transportation systems. NECTAR Conference, Delft, 1999.
3. Weiss G. Adaptation and Learning in Multi-Agent Systems: Some Remarks and a Bibliography. In Proceedings IJCAI'95 Workshop on Adaptation and Learning in Multi-Agent Systems. LNAI 1042, pp. 1-22, Springer, 1995.
4. Yuan, Y., Liang T.P., Zhang J.J. Using Agent Technology to Support Supply Chain Management: Potentials and Challenges. (available at: http://merc.mcmaster.ca/wpapers/wpaper456.html).
5. Stockheim T., Nimis J., Scholz T., Stehli M. How to Build a Multi-Multi-Agent System: The *Agent. Enterprise* Approach // In: Proceedings of the 6th International Conference on Enterprise Information Systems (ICEIS2004), 2004.
6. Nimis J., Stockheim T. The *Agent.Enterprise* Multi-Multi-Agent System // In: Special Track on Agent Technology in Business Applications (ATeBa04) at Multi-Conference on Business Information Systems (MKWI2004).
7. Zhengping L., Low M.Y.H., Kumar A. A Framework for Multi-agent System-Based Dynamic Supply Chain Coordination// In: Proceedings of the Workshop at the Second International Joint Conference on AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS (AAMAS 2003) (available at: http://www.agent-software.com.au/shared/resources/papers/aamas2003/aamas03w5-zhengping.pdf)
8. Fox M.S., Barbuceanu M., Teigen R. Agent-Oriented Supply-Chain Management // In: International Journal of Flexible Manufacturing Systems 12, 2000 –p. 165-188.
9. Henesey L.E. More than just Piers: A multi-agent systems on defining organization in a seaport terminal management system // In: Proceedings of 47th Annual Conference of the International Society for the Systems Sciences, Crete, Greece, 2003.
10. Gambardella, L.M., et al. The Use of simulation in the socio-economical evaluation of the management of an intermodal terminal. HMS 2000, Maritime & Industrial Logistics Modelling and Simulation, Portofino, Italy. (available at www.idsia.ch/~luca/hms2000.pdf).
11. Gambardella L.M, Rizzoli A.E., Zaffalon M. Simulation and Planning of the Intermodal Container Terminal. 1998. (available at www.idsia.ch/~luca/tr-idsia-41-98.pdf).
12. Gambardella L.M., Rizzoli E, Funk P. Agent-based Planning and Simulation of Combined Rail/Road Transport. (available at www.idsia.ch/~luca/simulation02.pdf).
13. Moore L.M., Reyns R.B., Kumara S.R.T., Hummel J.R. Distributed Intelligent agents for Logistics (DIAL). (available at ms.ie.org/SIW_LOG/97F/97F-SIW-112.rtf).
14. Reyns R.B., Kumara S.T.R., Satapathy G., Smith G., Hummel H.R. Distributed Intelligent agents for Logistics (DIAL). (available at ms.ie.org/SIW_LOG/98S/98S-SIW-203.rtf)
15. Funk P., Vierke G., Bürckert H.-J. Distributed Intermodal Transportation Planning. In: Multiagentensysteme in der Transportlogistic, 1999 (available at http://www.informatik.uni-bremen.de/agki/www/ki99-mas/funk_etal_99.pdf).
16. Sebastian H.-J, Nüßer G.H. Introduction to Minitrack: Intelligent Systems in Traffic and transportation // In: Proceedings of the 33rd Hawaii International Conference on System Science, 2000, -p.1-3.
17. Blumel E. (Ed.). Managing and Controlling Growing Harbour Terminals. Society for Computer Simulation International, 1997.
18. Bruzzone A.G., Signorile R. Genetic algorithms and simulation as support for planning a port terminal // In: Paper presented at Simulation in Industry, 8th European Simulation Symposium, Genoa, Italy, 1996.

19. Hayuth Y., Pollatschek M.A., Roll Y. Building a Port Simulator // In: Simulation, 63 (3), 1994-p. 179-189.
20. Rizzoli A.E., Gambardella L.M., Bontempi G. Simulation of an Intermodal Container Terminal to Assist the Management in the Decision Making Process, in proceedings of MODSIM 9, International Congress on Modeling and Simulation, Hobart, Tasmania, 1997.
21. Gerber A., Klusch M. Agent-Based Integrated Services for Timber Production and Sales. IEEE Intelligent Systems 17 (1), 2002-p. 33-39.
22. Bürckert H.-J., Fischer K., Vierke G. Holonic Fleet Scheduling with TELETRUCK // In: Proceedings of the Second International Conference on Computing Anticipatory Systems (CASYS'98), 1999.
23. Bürckert H.-J., Fischer K., Vierke G. Transportation Scheduling with Holonic MAS – the TELETRUCK Approach // In: H.S. Nwama and D.T. Ndumu (Eds.), Proceedings of the Third International Conference on Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM'98), 1998.
24. Bürckert H.-J., Vierke G. Simulated Trading Mechanismen für Speditionsübergreifende Transportplanung // In: H. Kopfer , C. Bierwirth (Eds.) Logistic Management – Intelligente I+K Technologien. Springer-Verlag, 1999.
25. Gerber C., Siekman J., Vierke G. Holonic Multi-Agent Systems. Research Report RR-99-03, DFKI, 1999.
26. Fischer K., Kuhn N. A DAI Approach to Modeling the Transportation domain. Research Report RR-93-25, DFKI, 1993.
27. Wooldrige M., Jennings N.R., Kinny D. The Gaia Methodology for Agent-Oriented Analysis and Design // In: Journal ofAutonomous agent and Multi-Agent Systems, 3(3), 2000 - p. 285-312.
28. Cossentino M., Potts C. A CASE tool supported methodology for the design of multi-agent systems. // In: Proceedings of the International Conference on Software Engineering Research and Practise, CREA Press, 2002.
29. Lind J. Iterative Software Engineering for Multi-agent Systems: the MASSIVE Method // In: Lecture Notes in Computer Science and Lecture notes in Artificial Intelligence, Vol. 1997, Sprimger-Verlag, 2001
30. Frey D., Stockheim T., Woelk P.-O., Zimmermann R. Integrated Multi-agent-based Supply Chain Management // In Proceedings of the 1st International Workshop on Agent-based Computing for Enterprise Collaboration, 2003.
31. Henoch J., Ulrich H. Agent –Based Management Systems in Logistics // In: Timm I.J. et al. (Eds.). 14th European Conference on Artificial Intelligence Workshop Notes, Agent Technologies and Their Application Scenarios in Logistics. 2000 - p. 11-15.
32. Moore L.M., Kumara S.R.T., Richbourg R.F. An Architecture for Logistics Replanning // In: Expert Systems with Applications, 11 (2), 1996 - p. 177-190.
33. Satapathy G., Kumara S.R.T., Moore L.M. Distributed Intelligent Agents for Logistics (DIAL) In: // Journal of Expert Systems Applications and Practice, 1998.
34. Gerber A., Vierke G., Zinnikus I. Generic Modeling of Multi-Agent Solutions for Transportation Domains // In: Timm I.J. et al. (Eds.). Proceedings of the ECAI 2000 Workshop 13 "Agent Technologies and Their Application Scenarios in Logistics", Berlin, 2000, pp. 29-33.
35. Scholder D., Eymann T. Technical Opinions: The Real Challenges of Mobile Agents // In: Communications of ACM 42 (6), 2000 – p. 111-112.

Vita Graudina, Riga Technical University, 1 Kalku str., Riga, LV-1658, assistant, M.sc.ing., vita.graudina@cs.rtu.lv

Janis Grundspenkis, Riga Technical University, 1 Kalku str., Riga, LV-1658, professor, Dr.habil.sc.ing., janis.grundspenkis@cs.rtu.lv

Graudiņa V., Grundspeņķis J. Aģentos sakņotas sistēmas, to arhitektūra un tehnoloģijas loģistikā.
Rakstā ir sniegts pārskats par esošajām aģentos sakņotām tehnoloģijām, kas domātas transporta un loģistikas problēmu risināšanai. Galvenā uzmanība ir pievērsta daudzaģentu sistēmu arhitektūrām un dažādiem aģentu tipiem tajās. Ir aplūkotas projektētās intelektuālas sistēmas, kas atbalsta piegādes ķēžu sistēmas projektēšanas fāzi (stratēģiskā plānošana) un/vai pārvaldības fāzi (taktiskā un operacionālā plānošana). Šāda veida sistēmas ir izstrādātas, balstoties uz moderniem programmatūras inženierijas konceptiem, tajā skaitā objektorientēto programmatūras izstrādi un intelektuālu aģentu paradigmu. Tradicionāli aģentos sakņotas sistēmās transportā un loģistikā tiek izmantotas transporta plūsmu modelēšanai un imitēšanai, loģistikas plānošanai, piegādes ķēžu pārvaldībai, lēmumu pieņemšanas atbalstam, jo šie uzdevumi ar lielo iesaistīto pušu skaitu, to dažādību un mainību, ir atbilstoši intelektuālu aģentu būtībai. Rakstā aplūkotas arī jaunākās tehnoloģijas, kas balstās uz aģentu paradigmu, tās ir multi-multi- aģentu sistēmas un holonu aģenti. Raksts sniedza arī ieskatu aģentos sakņotu sistēmu izstrādes metodoloģijās. Rakstā galvenā uzmanība vērsta uz aģentos sakņotām sistēmām, to arhitektūru un tehnoloģijām. Šobrīd ir izstrādātas vismaz divas dažādas aģentos sakņotas arhitektūras, precīzāk, daudzaģentu arhitektūras, un tiek lietotas reālās transportēšanas un loģistikas problēmās. Ir aprakstīti divi labi zināmi projekti – TELETRUCK (un tā modifikācijas) un DIAL.

Graudina V., Grundspenkis J. Agent-Based Systems, Their Architecture and Technologies from Logistics Perspective.
This paper is an overview of the proposed agent-based technologies for problem solving in transport and logistics domain. Main attention is paid on already proposed multi-agent system architectures and different agent types used in these architectures. Intelligent systems designed to assist the design-phase (strategic planning) of traffic and transportation systems and/or the management phase (tactical and operational planning) are described. They are built on the bases of advanced software engineering concepts including object-oriented software development and intelligent agent paradigm. Traditionally agent-based systems for transportation and logistic are used for traffic simulation and modelling, logistics planning, supply chain management, decision making support systems because those tasks with many involved organizations, means, their variety and continuously changing, are appropriate to intelligent agents. The papers shows also newest technologies based on agents paradigm, that are multi-multi-agent systems and holonic agents. The paper gives small overview of methodologies for agent-based system development. The paper is focused on agent based systems, their architectures and technologies. At least two different agent based architectures or to be more precise, multi-agent architectures has been worked out and used in real life problems within the transportation and logistics domain. Two well-known projects, namely, TELETRUCK and its modifications, as well as DIAL are described in more details.

Граудиня В., Грундспенькис Я. Системы основанные на агентах, их архитектуры и технологии в логистике.
Статья является обзором предложенных технологий интеллектуальных агентов для решения проблем в области транспорта и логистки. Главное внимание уделено известным типам архитектур интеллектуальных мульти-агентных систем и различным типам агентов, используемых в этих архитектурах. Описаны интеллектуальные системы, которые поддерживают проектную фазу (стратегическое планирование) транспортных систем и/или фазу управления (тактическое и операционное планирование). Такие системы строятся, используя современные концепции программной инженерии, включая интеллектуальных агентов и объектно-ориентированную разработку программного обеспечения. Традиционно системы на основе агентов в области транспорта и логистики используются для имитации и моделирования транспортных потоков, планирования логистики, управления цепями поставок и поддержки принятия решений, потому что эти задачи с большим числом вовлеченных сторон, их разнообразием и изменчивостью, соответсвуют сущности интеллектуальных агентов. В статье описаны новейшие технологии, основанные на агентах: мульти-мульти-агентные системы и холоник агенты. Статья содержит и введение в методологии для разработки агентных систем. В статье главное внимание уделено системам, основанным на агентах, их архитектурах и технологиях. Анализ литературы показал, что разработаны по крайней мере две мульти-агентные архитектуры, которые используются в решении реальных проблем в области транспорта и логистики. Два известных проекта - TELETRUCK и его модификации, и DIAL рассмотрены более детально.